Html : HTML genel olarak bilinenin aksine bir programlama dili değilidir. Html açılımı hiper metin işaretleme dili(Hyper Text Markup Language) olan bir metin işaretleme dilidir. Bu dile ile oluşturulan metinler chrome, Explorer gibi tarayıcalar tarafından okunarak kullanıcıya gösterilir. Bir web sayfasının barındırdıklarını ve genel hatlarını barındırır.1980 yılında CERN’de çalışanların birbirileri ile dosya ve veri paylaşmak amacı ile ilk taslağı geliştşrilmiştir. 1993 yılında oluşturulmuştur ve günümüze kadar çeşitli geliştirmeler ile gelerek hala yaygın bir şekilde kullanılmaktadır.

HTML ne işe yarar : Sahip olduğu etiketler yardımı ile web sayfasındaki metinlerin nasıl okunacağını belirler. Böylece düz metinlerimiz kendisine verilen etiketin özelliklerini alarak istediğimiz şekilde görünmüş olur.

Html kullanım alanları :

* Web projeleri
* dökümantasyon

Css: İnternet kullanımın giderek artması ile HTML in arayüzü yeterli görülmemiş ve bu nedenle arayüz konusunda daha geniş bir yelpazeye sahip olan css geliştirilmiştir Css, basamaklanmış stil katmanları anlamına gelen Cascading Style Sheets’ in kısaltılmış harfleridir. Css W3C tarafından 17 aralık 1996 da css1 sürümü ile yayınlanmıştır. Her yapılan geliştirme css 1 in üzerine inşa edilerek zamanla 1998 yılında Css2 ve son olarak Css’in son sürümü 1998 yılının haziran ayında Css3 çıkmıştır. Css 3 farklı şekilde html sayfalarına dahil edilebilir. Bunlar;

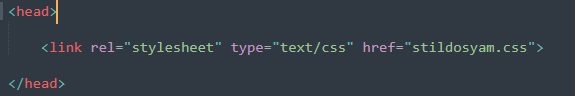
* Satır içine ekleme
* Dahili ekleme
* Harici ekleme

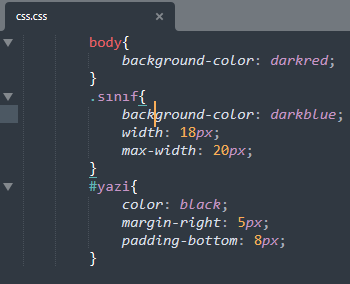
Dahili ekleme: Css Kodlarını dahili şekilde eklediğimiz zaman bu eklenen kodlar yalnızca bu sayfa için geçerli olur. Tekrar kullanılamadığı için geliştiriciye gereksiz iş yüküne sebep olabilir. Ancak özel olarak tasarımı ayrı olan sayfalar için dahili ekleme yöntemi kullanılabilir.

Html etiketlerinden olan head etiketlerinini arasına style etiketi açılarak css kodları yazılır ve yine style kapatma etiketi ile kapatılarak tanımlanmış olur.



Harici ekleme : bu yöntemde css kodları uzantısı .css olan bir dosyaya yazılarak dosya kayıt edilir. Daha sonra oluşturulan html sayfasına eklenmek istenen harici css dosyası head etiketleri arasındalink etiketi ile rel özelliği stylesheet olarak tanımlanarak href özelliği ile dosya yolu verilerek dosyaya dahil edilmiş olur. Böylece yazılanlar tekrar kullanılabilir ve bütün bir sayfanın yapısına entegre edilmiş olur.





Satır içi stil ekleme: Satır içi stil eklemede stil yalnızca bir öğe için geçerli olur. Bu sebebp ile her öğen için istenilen stilin tekrar yazılması gerekir. Satır içistil ekleme, stilin ekleneceği html öğesinin style özelliğinin içerisine css kodlarının yazılması ile yapılır.



CSS ne işe yarar:Csskodları html öğelerinin yazılan kodlara göre görünüşlerinin değiştirilmesini sağlar. Böylece kullanışlı ve göze hitap eden sayfalar oluşturmuş oluruz.

CSS kullanım alanları :

* Web sayfaları

Javascript: java script Brendan eich tarafından 4 aralık 1995 yılında çıkarılmıştır. Yazılım geliştiriceli tarafından uygulama, web sayfaları ve sunucu geliştirirken kullanılan bir programlama dilidir. Hafif ve dinamik olması java scriptin önemli özelliklerindendir. Günümüzde popüler olan birçok tarayıcı tarafından desteklenmektedir.bir çok framework ve javascript kütüphanesine sahip olan bu dil tek başına programlama yapmaktan ziyade küçük scriptler yazılarak web sayfalarının etkileşimini sağlar. Hafif ve hızlı bir programlama dilidir.

JS ne işe yarar : Java Script dinamik ve etkileşimli web sayfaları oluşturulmasını sağlar. html ve css ile etkileşime girerek canlı halde türlü değişiklikler yaılmasına olanak sağlar.

Js Kullanım alanları: Web suncusu

* Web sayfaları
* Suncu uygulamaları
* Web uygulamaları
* Mobil uygulamalar
* Oyun geliştirme

Java: Java James goslin tarafından tasarlanan ve oracle tarafından geliştirilen bir programlama dilidir. İlk defa 23 mayıs 1995 de yayımlanmış olup hala günümüzde popülerliğini korumaktadır. Açık kaynak kodlu nesneye yönelik ve bağımsız platform bir dildir. Yüksek seviye bir dilolup anlaşılması ve okunması kolaydır.

Ne işe yarar : büyük çaplı projelerden küçük çaplı projelere her projede verişmli olan bu dil istemci tabanlı bir çok uygulamanın geliştirilmesinde görev alır. Nesne tabanlı olan bu dil gerçek dünya problemlerini bilgisayar ortamında çözülmesini sağlar.

Kullanım alanları:

* Oyun geliştirme
* Bilgi bulut işlem
* Büyük veri
* Yapay zeka
* Nesnelerin interneti
* Web uygulamaları
* Mobil programlama
* Veri analitiği
* Gömülü ssitemler
* Bilgisiaiyar uygulamaları
* Eğitim

Java ve Javascript farkları:

* Java Daha büyük ve karmaşık projelerde kullanılırken javascript daha küçük scriptler yazılmada kullanılır.
* Java nesne tabanlı bir programlama dili iken java script daha çok betik bir dildir.
* Java script web sayfalarını yalnızca dinamik ve etkileşimli hale getirmede kullanılırken java aynı zamanda sunucu tarafı uygulamaları ve bağımsız programlarda kullanılşır
* Java kodların yeniden kujllanımını sağlar ve sınıflar ve nesneler ile daha derli bir düzen sunar iken javascriptin buna desteği yoktur.
* Java kalıtım, veri kapülleme ve polimorfizm gibi birçok özelliği destekleyebilir.
* Java birçok platforma çalışabiliriken java script yalnızca tarayıcılarda çalışır.
* Java derlenirken java script yorumlanır

Typescript : Typescript Microsoft tarafından hazırlanan ilk defa 1 ekim 2012 ‘ de yayımlanan bir programlama dilidir. Bünyesindeki derleyici ile yazılan typescript kodlarını java scripte çevirir ve çıktı olarak Alınan java script kodları kullanılır. Type script’i ister sunucu istersek de istemci tarafında kullanabiliriz. Java scripitin sahip olduğu bütün özellikeleri barındıran TypeScript aynı zamanda yeni eklenen özellikler ile daha kullanışlı bir programlama dili olmayı başarmıştır. Büyük ve daha karmaşık projelerde Javascripte göre daha kullanışlı, okunabilir, geliştirmeye müsait ve verimli bir sonuç çıkartır. TypeScript'i JavaScript'e derlemek için ya varsayılan TypeScript Checker kullanılabilir, ya da Babel derleyicisi çağrılabilir(wikipedi).

Ne işe yarar: Javascriptin yaptığı hemen hemen her işi yapan bu teknoloji web sayfalarının dinamik bir halde etkileşimleştirilmesinde kullanılır.

ts Kullanım alanları: Web suncusu

* + - * Web sayfaları
      * Suncu uygulamaları
      * Web uygulamaları
      * Mobil uygulamalar
      * Oyun geliştirme

Angular: Angular Google tarafından geliştirilerek ilk defa 14 eylül 2016 tarihinde yayımlanmıştır. Hala geliştirilme süreci devam eden bu teknoloji sürekli güncelleme almaktadır. Çapraz platform olan angular bu özelliği sayesinde bir çok platformda çalışabilmektedir. Angular SPA yapısını desteklemektedir. İngilizcesi Simple Project Application olan Spa nın Türkçe karşılığı tekil projedir. Yapısında MVC yani Model view controller mimarisini barındırır. Bu sayede daha kullanışlı okuması ve yazması daha kolay ve geliştirmeye açık bir yapı sunmuş olur.

Ne işe yarar: yeni bir sayfa yüklemek yerine yeni sayfanın öğerlerini dinamik olarak yazarak hazır hale getirilmesini sağlar. Type script ile yazılan bu teknolloji web sayfaları geliştirmede kullanılır.

Kullanım alanları:

* Web sayfaları
* web uygulamaları

Springboot : Springboot java programlama diline ait spring framework ile geliştirilmiştir. Kendisi bir framework değildir yalnızca spring frameworkünün xml yapıları ile uğraşmamamızı sağlayan bir yapıdır. Java programlama dilinde microservis yazılmasını sağlar. Spring framework ile geliştrilen uygulamalar bir çok özel ayara ihtiyaç duyar. Bunun sebebi spring framework’ün çok geniş bir geliştirme yelpazesinin olmasıdır. İçerisinde birçok kütüphane barındıran bu framework bazen her uygulama için standart ayarlamaları gerektirirken çok karmaşık ve yorucu olabiliyor. Springboot ise sizin için bu ön ayarlamaları varsayılan olarak yaparak projemizi ayağa kaldırıyor. Ancak sizin yapacağınzı değişiklikler var ise ozaman müdahele ederek istediğiniz yeri projenize göre ayarlayabilrsiniz. Böylece geliştrirciyi birçok iş yükünden kurtarır iken kullanışlı projeler ortaya koymamıza olanak tanıyor. İçerisinde barındırdığı anatasyonlar ile geliştiricileri büyük bir yükten kurtarır. Anotasyonlar uygulamanın davranışını belirler. Springboot geliştiricinin en stabil bir programı hızlı ve uygun yöntemler ile geliştirmesinin sağlamaktır. Bu bağlamda sahip olduğu anotasyonlar bir projede bir çok yerde kullanmamız gerekecek bazı kod öbeklerinin tek bir çağırma işlemi ile uygulama davranışını en uygun şekilde belirler. Mvc mimarisini kullanan springboot bu sayede kullanışlı ve geliştirilmeye açık projeler ortaya çıkmasını sağlar. Springboot spring frameworkün sahip olduğu Dependecy injection özelliğini barındırır bu sayede bir nesnenin ihtiyacı olan ve bir deyime bağlı olan nesneleri tekrar oluşturmak yerine dışarıdan enjekte eder. Böylece bizi yine zamandan tasarruf etmeye, oluşabilecek hataların önüne geçmeye, ve kod tekrarından yani iş yükünden kurtaracaktır.

Springboot neişe yarar: Springboot mikroservisler yazabilir, apiler oluşturabiliriz. Yani web tabanlı uygulamaların backend tarafını geliştirebiliriz. Geliştirilen microservis ile kötü amaçlı yazılımlara karşı güvenlik önlemleri alarak sistemimizi daha güvenli hale getirebiliriz. Front end tarafından yapılan istekleri yakalayarak bunlara gerekli şekillerde cevap verebiliriz. Veri tabanı ile bağlantı kurarak verileri çekebilir ve java nesnesine dönüştürerek hata payını en aza indirip modeller yardımı ile kullanabiliriz. Sahip olduğu DTO (Data Transfer Object) yapısı ile veri fazlalığından kurtularak sistemdeki yükü hafifletebiliriz.Geliştirilen microservisler sayesinde sistemimiz daha hızlı ve güvenli olacaktır. Yani kısaca Springboot ile geliştirilen uygulamalar sayesinde veri tabanı ile bağlantı kurabilir veritabanından elde ettiğimiz verileri modeller yardımı ile java nesnesine çevirerek view katmanında gösterilmeye hazır hale getirebiliriz.

Spring boot nerelerde kullanılır:

* Yapay zeka
* Bulut sistemler
* Veri depolama
* Veri analizi
* Nesnelerin interneti
* Güvenlik ve idare
* Api Gateway oluşturma
* Mikro hizmetler
* Web tabanlı projeler

Sql : Sql, ingilizcesi Structed Query Language olan Yapılandırılmış sorgu dili’ nin kısaltılmış halidir. Sql verilerin Farklı boyutlarda farklı yapılarda saklayabilen yapılan sorgulara cevap verebilen bir veritabanı uygulamasıdır. SQL sayesşnde bu verilerin yönetilmesi, silinmesi,aktif edilmesi, güncellenmesi sağlanır. Çıktığı ilk günden bu yana popülerliğini koruyan hala yapısı ile birçok projede yeterlilik gösteren Sql günümüzde en popüler veri tabanlarından ve sorgu dillerinden birtanesidir. Sql genel bilinenin aksine bir programlama dili olmamasına karşın veri tabanında kullanılan bir alt dildir. Bazı Sql dilinin kullanıldığı veritabanları aşağıda listelenmiştir:

* MySQL
* Mssql
* Microsoft SQL Server
* Oracle
* IBM Database 2 (IBM DB2)
* PostgreSQL
* Sybase
* IBM Informix
* Progress
* Firebird
* Access

SQL ne işe Yarar: SQL sayesşnde verilerin yönetilmesi, silinmesi,aktif edilmesi, güncellenmesi sağlanır. Barındırdığı sorgu yöntemleri ile veri tabanında tablo oluşturma tablo bağlama stün satır seçme birleştirme gibi birçok veri tabani yönetim yöntemlerini kullanmamızı sağlar. Sql dili sayesinde birden fazla tablolardaki satır ve stunları hücresinde barındırdığı veriye göre filtreleyebilir ve bu filtre akabinde seçim, silme, güncelleme, veya kayıt etme işlemi yapabiliriz.

Sql Kullanım alanları:

* Web tabanlı uygulamalar
* Veri Depolama
* veri analizi
* Windows form uygulamaları
* oyun geliştirme
* Yapayzeka
* Ve veri yönetmininin olduğu her alanda sql kullanılabilir.

Primeng: Türk mühendisler Tarafından kurulan ve geliştirilen prime teknolojilerinin 2009 yılında ortaya çıkardığı PrimeFaces kütüphanesinin Angular için Oluşturulmuş halidir. Primeng Front end tarafında kullanılan html,css ve javascript teknolojilerinin kalıplaşmış kodlarının tekrar tekrar yazılmasından bizi kurtararak zamandan ve iş yükünden tasarruf etmemizi sağlar. Anlaşılacağı üzere bir angular kütüphanesidir.

Ne işe yarar: Primeng etkiketlerini kullandığımız zaman html için öncesinde yazılmış özellikleri dahil bir şekilde kullanabiliriz. Aynısı cs ve javascript içinde geçerlidir. Böylece her projede yazdığımız artık standartlaşmış bazı kalıpların iş yükünden bizi kurtarırken daha düüzenli, kullanışlı ve göze hitap eden tasarımlar ortaya çıkarmamızda yardımcı olur.

Kullanım alanları:

* web tabanlı projeler
* Angular
* Vue
* Reactjs

Bootstrap: Bootstrap Html Css ve Javascript Kullanılarak oluşturulmuş bir Frameworktür. Geliştiricilerin ve tasarımcıların Zmandan ve iş yükünden tasarruf ederek etkileşimli web siteleri yazmalarına olanak tanır. Bootstrap frameworkü Twitterın eski çalışanlarından Mark otto ve JAcobthornton tarafından geliştirilmiştir. İlk defa 19 ağustos 2011 tarihinde yayımlanan bootstrap web proramlama alanında kullanılan en yaygın frameworklerden birisidir.

Ne işe yarar: Bootstrap css ve javascript teknolojilerinden yararlanarak oluşturulmuş özellikleri tek bir çatı altında bize sunar. Böylece elimizde tekrar kullanılabilir bir kod paketi olur. Bize zamandan ve iş yükünden tasarruf etmemize olanak sağlar. Boostrap geliştirme aracında tanımlanan özellikler html öğelerinin “class” özelliğine tanımlanarak framewörkten çağırılabilir ve tanımlı özellikler kullanılabilir. Eğer Kullanılan paketin içeriğinde olmasını istemediğimiz bir özellik var ise owerride edilerek düzeltilebilir. Baskın gelmesi istenilen özellik html öğesinin style özelliğinin içerisine css kodları ile owerride edilebilir.

Monolitik mimari: Monolitik mimari bir uygulamanın tüm parçalarının tek bir çatı altında sunulması olarak tanımlayabiliriz. Geliştirilecek olan projenin az kaynak tüketmesi, zamandan ve insan gücünden tasarrufu için küçük çaplı projelerde monolitik mimariyi kullanmak faydalı olacaktır. Böylece geliştirme kısmı hızlıca tamamlanarak proje kaynaklarından tasarruf edilerek ürün en hızlı şekilde ortaya çıkarılmış olur. Ancak geliştirme ve kullanım safhasında geniş kitleler ile ve yoğun istekler ile karşıkarlıya kalındığında monolitik mimari yetersiz kalabilir. Bu neden ile küçük çaplı ve az karmaşık projelerde tercih edilir. Küçük çaplı ekiplerin yönetilmesi ve uygulama geliştirmesi kolaydır. Birden fazla ekibe ihtiyaç duymaz ve birden fazla modülle çalışmamıza olanak sağlar.

Yazılım tek bir kod tabanında olduğundan dolayı güvenlik endişesi düşüktür. Ağ gecikmesi çok yapılı mimarilere göre daha azdır ve yapı tek çatı altında olduğundan dolayı hataları bulmak ve veri akışını izlemek daha kolaydır.

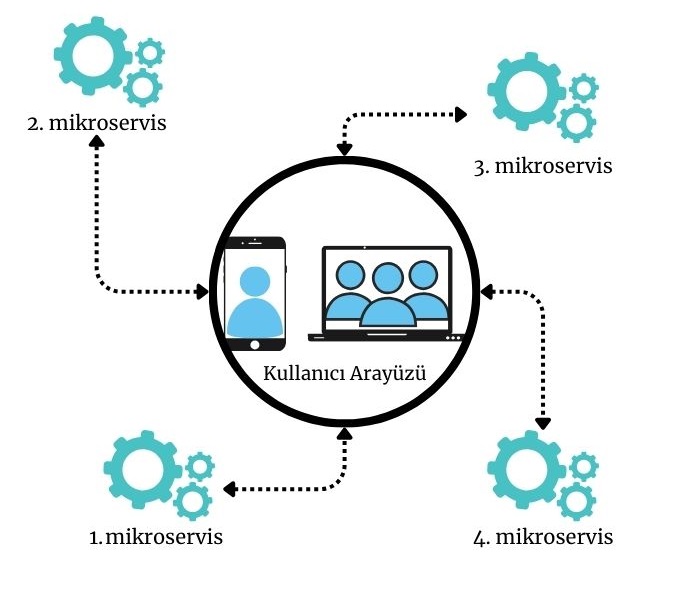
Monolitik mimari ne işe yarar: monolitik mimari kaynaklardan tasarruf ile elimizde olan az miktardaki kaynakları kullanarak projeler geliştirmeye olanak sağlar. Bir projenin gereksinim duyduğu herşeyi bir çatı altında toplayarak amacını gerçekleştirmeye çalışır.

Monolitik mimarinin kullanım alanları :

* Web tabanlı projeler
* Küçük çaplı projeler
* Az karmaşık projeler

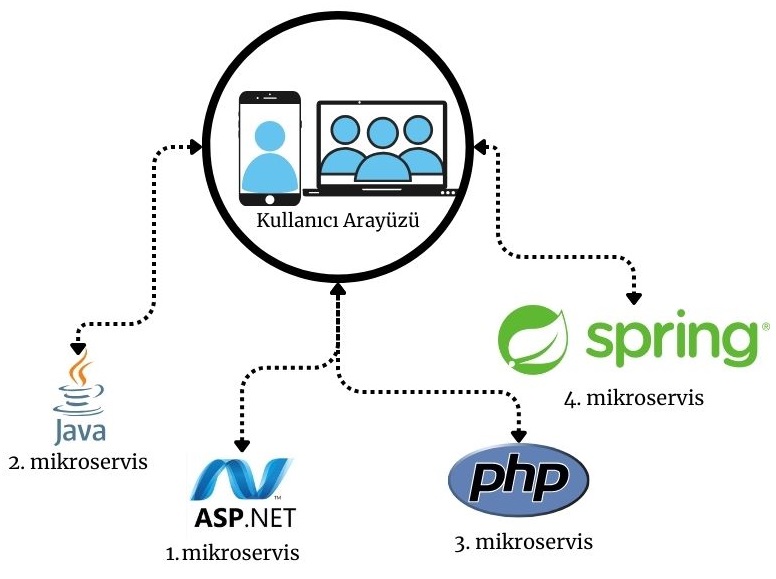
Microservis: Mikroservis mimarisi monolitik yapının yeterli gelmediği yerlerde geliştiricileri büyük çaplı projeleri küçük servislere bölme ihtiyacı ile gelişmiştir. Öncesinde kullanılan monolitik mimari barındırdığı dezavantajlardan dolayı yetersiz gelmeye başlamış ve geliştiricileri yeni arayışlara yöneltmiştir. Bu sayede büyük bir ekip ile büyük ve karmaşık bir projeyi tek çatı altında geliştirmektense bu projeyi küçük mikroservislere bölerek ve yönetilmesi zor olan büyük ekipleride parçalayarak her ekibe faklı bir görevler verilmiştir. böylece mikro servis parçaları yönetilmesi kolay ekipler tarafından geliştirilerek belirlenen bazı standartlar doğrultusunda senkron veya asenkron olarak çalışan bir hizmet çıkarmıştır.

Birden fazla teknolojiyi bir arada kullanmamıza olanak sağlayan mikroservis mimarisi bu özelliği ile geliştirilen projelerin daha yenilikçi, değiştirilebilir ve geliştirilebilir olmasını sağlamıştır.



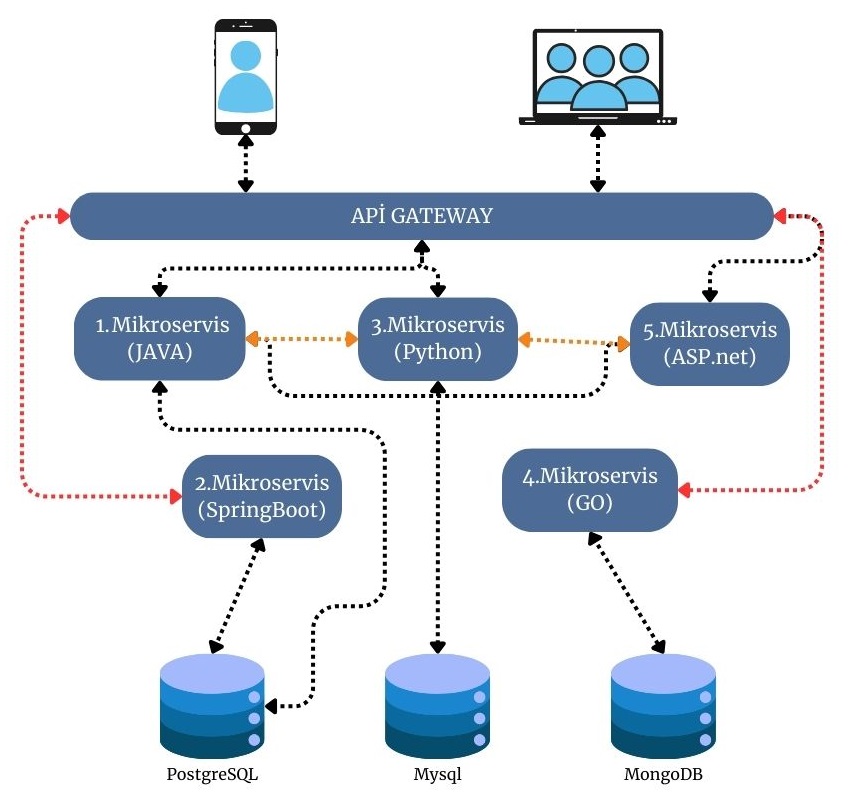
Farklı yapıların bir araya gelmesiye oluşan bu sistem, ortaya çıkan bir arızanın bütün sistemi etkilemesinin önüne geçmiştir. Böylece arıza sadece bağlı olduğu servisleri etkileyerek büyük çaplı hatalardan geliştiricileri kurtarmıştır. Geliştirilmesi ve değiştirilmesi kolay olan bu sistemde arızalar küçük çaplı olduğu için arıza çözümleride kolay ve hızlıdır. Yine farklı yapıların bir araya gelmesi ile oluşturulduğu için sisteme yeni bir sistem entegre etmek veyea entegre bir sistemi çıkartmak monolitik yapıya göre oldukça kolaydır. Sistemin tamamı ile değil sadece düzenleme yapılacak olan hizmeti ile ilgilenilir. Böylece kolayca ekleme kaldırma ve değişiklik yapılabilir. Düzenleme yapılacak hizmet sistemden çıkarılsa dahi sistemin geriye kalan kısmı çalışmaya ve hizmet vermeye devam eder.

Mikroservis mimarisi ne işe yarar : Mikroservis mimarisi yapılacak projeyi birden fazla mikro hizmetlere bölerek kullanıcının client kısmında yaptığı her isteğe bir başka mikroservisin cevap vermesini sağlar. Yine kullanıcının her isteği uygulama tarafından karşılanır sadece her istek farklı bir hizmet tarafından farklı bir veritabanından çekilerek karşılanabilir. Böylece büyük projemiz küçük mikro hizmetlere bölünmüş olur. Bunu sağlar iken her mikro hizmeti o işi en iyi şekilde yapcak teknolojiyi kullanrak yazabilirz.



Mikroservis mimarisinde servisler arası haberleşmeyi ve servislerin client ile haberleşmesini yani talep göndermek ve gönderilen talepleri karşılamak için http veya gRPC gibi bazı protokoller kullanılır. Kullanıcı uygulamada bulunan geliştiricilerin öncesinde belirlediği bir şekilde etkileşime girerek sunucu client tarafından bir http ile bir talep gönderir. Sunucu bu talebi yine http protocolünü kullanrak alır ve gerekli olan işlemleri yaparak bilgiyi hazır hale getirir. Hazır hale gelen bilgi yine http protocolü ile sunucu tarafından client tarafına gönderilir. Client kendisine gelen bilgiyi alarak gösterilmesini istediğimiz şekilde kullanıcıya sunar.

Mikro hizmetlerin client ile haberleşmesini sağlarken en sık kullanılan yöntem bir api geliştirmektir. Yazdığımız mikroservise bir api gateway geliştirerek mikroservisin hangi istekleri nasıl karşılayacağını belirleyebiliriz. Böylece bu apiye uygun gelen her istek hangi client’dan gelirse gelsin buna uygun bir şekilde karşılayabilir ve verileri yine aynı protokol ile sunabiliriz. Burada yaptığımız şey aslında bir standart belirlemek. Mikro hizmetimize giriş ve çıkış standartı oluşturuyoruz. Bu standartın ne olduğunu bildikten sonra uygulamamız, her platformda uygulama geliştirilebilir hale geliyor.



MONOLİTİK MİMARİ VE MİKROSERVİS FARKLARI

Api: API, ingilizcesi “Application Programming Interface” olan uygulama programlama arayüzü kelimelerinin kısaltılmış halidir. API terimi, ilk defa 1974 yılında Christopher J. Date tarafından yayımlanan "The Relational and Network Approaches: Comparison of the Application Programming Interface" adlı makalede geçmiştir(***https://coderspace.io/sozluk/api***). Yazılım uygulamalarının birbirileri ile iletişim kurmasını sağlayan bir arayüzdür. APIler bir programın bazı işlevselliklerini diğer programlar tarfından erişilmesi ve kullanılması sağlar. Programlar arası iletişimi kurarak aralarında bir veri yolu oluşturur. Böylece farklı teknolojilerdede olsalar birçok yazılım programlarının birileriyle entegre edilmesini ve veri alışverişini sağlayarak birlikte bir işbirliği içerisinde olmasına imkan tanır.

Apiler birçok farklı türde ve teknolojide geliştirilebilir. Örneğin web apiler web tabanlı uygulamalar arası veri iletişimini kolaylaştırır. Web sitelerinin veri tabanlarına ulaşmasını sağlar. Farklı şekillerde üretilmişlerde olsa farklı teknolojilerdeki web sitelerini Web API Keyler sayesinde birbirilerine kolayca entegre olabilirler. Böylece Bir web sitesi bir sunucudan hava durumu bilgilerini alırken başka bir sunucudan altın, döviz, borsa gibi bilgileri alarak başka bir sunucu ile ödeme işlemleri yapabilir.

Apiler HMTL, XML,JSON gibi protocolleri kullanırlar, bu sayede farklı sistemlerin birilerine entegrasyonunda veri alışverişi için bilginin geldiği formatı bilmek yeterlidir. Böylece Birbirlerinin hangi teknolojilerde nasıl geliştirildikleri kadar detaylı bilgilere ihtiyaç duymadan sadece api keyleri kullanrak kolayca entegre edebilir ve veri akışını sağlayabiliriz. Verdiğimiz örnekler web API üzerinde olsa da API ler işletim sistemleri, veri tabanları, gömülü sistemler gibi bir çok alanda kullanılabilirler.

API ler sistemlerin birbirilerine entegre olmalarında kolaylık sağlarlar. Böylece iki farklı sistemde herhangi bir değişikliğe gerek duymadan sistemler birbirilerine entegre olabilirler. Bu da geliştiricilere kaynakları daha verimli kullanarak zamandan, ücretten, iş yükünden tasarruf etmeyi sağlar. ***https://www.linkedin.com/pulse/api-g%C3%BCvenli%C4%9Fi-temelleri-m%C3%BCcahit-i%CC%87%C3%A7/?originalSubdomain=tr***

Apilerin amacı iletişim sağlamak olduğu için iletişim kurmak isteyen sistemlere karşı güvenlik önlemleri almak gerekmektedir. Zamanla api için birçok güvenlik yöntemi geliştirilmiştir. Bunlardan iki yetkilendirme kimlik doğrulamadır. Veri güvenliğini sağlamak için öncelikle sistemimize giren kişinin kimliğini öğrenmeli ve giriş iznin olup olmadığını kontrol etmeliyiz. Eğer giriş izni var ise sırada verilerimizden hangilerine erişebiliyor bunun konytolünün sağlanası gerekir. Bu noktada yetkilendirme yapmak ve yetkisi dahilindeki verileri sunmalıyız.

Veri alışverişi sırasında verilerimizin güvenliği için veri şifreleme yöntemleri kullanılır. Taleplerimizde kullandığımız http methodunu hassas bilgileri saklamak için https yani http Secure protocolünü kullanabiliriz. Böylece verilerimizi iletişim esnasında güvence altına almış oluruz.

İstek sınırlama: İstek sınırlama sistemimize çok sayıda istek göndererek kaynaklarımızı doldurmayı amaçlayan kötü niyetli saldırıların hizmet reddi ile önlenmesi kaynaklarımızın doğru kullanımını veri güvenliğini ve kullanıcı memnuniyetini arttıracaktır.

Bunlar gibi günümüze kadar karşılaşılan her soruna üretilmiş birçok hata önleme ve güvenlik önlemleri bulunmaktadır.

Mimari Yapılarına göre aPI ler

APIler mimari yapılarına göre sınıflandırılabilir. Bunlar;

REST API: REST API, ingilizcesi “Representational State Transfer” olan Temsili durum aktarımı kelimelerinin kısaltılmış halidir. http protocolünü kullanan bir api servisidir. http protocolünün delete, put, post, get işlevlerini kullanarak çalışır. Veri alış verişinde XML veya diğer fotmatlar yerine JavaScript Object Notation (JSON) veri formatını kullanır.

REST API, yeni nesil web tabanlı projelerin geliştirilmesi için sıklıkla kullanılan bir mimari haline gelmiştir. İstenilen hemen her ihtiyacı karşılaması, bakım kolaylığı, diğer sistemlere entegre edilmesinin kolay bir mimari olması ve hızlı, hafif bir yapıya sahip olması günümüzde web tabanlı uygulamalarda en çok tercih edilen mimarilerden birisi olmasını sağlıyor.

SOAP: SOAP, ingilizcesi Simple Object Access Protocol olan basit nesne erişim protocolü anlamına gelen kelimelerin baş harfleridir. Apiler içerisinde sıkı bir güvenlik yapısına sahip olması ile bilinir. Soap veri transferinde XML formatını kullanarak iletişim kurar. Rest api ye göre yapılandırması zor olsada sağladığı güçlü güvenlik yapısı kendisni büyük ve güvenliğin ilk planda bulunduğu projelerde sıklıkla başvurulan bir mimari haline getirmiştir.

Postgresql: İsmi konusunda, başlarda Kaliforniya Üniversitesi'nde geliştirilen Ingres veritabanının devamı niteliğinde olduğu için bu isme atıfta bulunularak POSTGRES adını aldı ***.(***[***^***](https://tr.wikipedia.org/wiki/PostgreSQL#cite_ref-design_4-0)***Stonebraker, M.; Rowe, L. A. (May 1986).***[***The design of POSTGRES***](http://db.cs.berkeley.edu/papers/ERL-M85-95.pdf)***(PDF). Proc. 1986 ACM***[***SIGMOD***](https://tr.wikipedia.org/w/index.php?title=SIGMOD&action=edit&redlink=1)***Conference on Management of Data. Washington, DC. 6 Aralık 2019 tarihinde kaynağından***[***arşivlendi***](https://web.archive.org/web/20191206194718/http:/db.cs.berkeley.edu/papers/ERL-M85-95.pdf)***(PDF). Erişim tarihi: 6 Mart 2021.***

1. [***^***](https://tr.wikipedia.org/wiki/PostgreSQL#cite_ref-about/history_5-0)[***"PostgreSQL: History"***](https://web.archive.org/web/20170326020245/https:/www.postgresql.org/about/history/)***. PostgreSQL Global Development Group. 26 Mart 2017 tarihinde***[***kaynağından***](https://www.postgresql.org/about/history/)***arşivlendi.***

***)***

1996 yılında SQL desteğini vurgulamak maksatlı PostgreSQL şeklinde değiştirildi. Son olarak 2007 yılında isim değişikliği konusu topluluğun tartışmasına açıldı ve geliştirme ekibi PostgreSQL isminin devam etmesine, Postgres isminin de takma isim olarak kalmasına karar verdi. ([***^***](https://tr.wikipedia.org/wiki/PostgreSQL#cite_ref-Project_name_6-0)[***"Project name – statement from the core team"***](https://web.archive.org/web/20071118121135/http:/archives.postgresql.org/pgsql-advocacy/2007-11/msg00109.php)**. archives.postgresql.org. 16 Kasım 2007. 18 Kasım 2007 tarihinde**[***kaynağından***](http://archives.postgresql.org/pgsql-advocacy/2007-11/msg00109.php)**arşivlendi.**)

POSTGRESQL hem ilişkisel Hemde ilişkisel olmayan veri yapısını destekleyen bir sistemdir. Açık kaynaklı kullanımı ücretsiz olan bu veri tabanı kurumsal sınıfı desteklediği gibi gelişmiş bir nesne ilişkisel bir veri tabanıdır. SQL desteği sunarak karmaşık verileri ve akabinde büyük iş yüklerini yüksek performans sergileyerek işler.

Windows, macOS, linux, UNIX gibi günümüzde polülerliğini koruyan işletim sistemleriyle uyumludur. Javascript, Python, java, C#, C, C++ gibi güncel dillere destek vermektedir. Primivites, Structured, geometry, document gibi bir çok veri tipini bünyesinde saklayabilir ve işleyebilir. Yabancı anahtarlar, birincil anahtarlar, dışlama kısıtı, tavsiye kilitleri, açık kilitler, NOT NULL ve UNIQUE gibi özeliikleri bünyesinde barındırır ve bu sayede veri bütünlüğünü korur. LDAP, SSPI, GSSAPI gibi birçok kimlik doğrulama protokollerini kullanır. Çok faktörlü kimlik doğrulama gibi birçok güvenlik önlemi alarak barındırdığı veriler için güvenli bir ortam sağlar.

PostgreSQL’in özellikleri aşağıdaki tabloda verilniştir.

|  |  |
| --- | --- |
|  |  |
| **Veri Tipleri** | **Primitives:** Number (int), Numeric(int, float), String ve Boolean  **Structured:**Date/Time, Array, Range, UUID  **Document:** JSON/JSONB, XML, Key-value (Hstore)  **Geometry:** Polygon, Point, Circle, Line  **Customizations:**Custom Types and Composite types |
| **Veri Bütünlüğü** | UNIQUE, NOT NULL  Primary Keys (Birincil Anahtar)  Foreign Keys (Yabancı Anahtarlar)  Exclusion Constraints (Dışlama Kısıtı)  Explicit Locks, Advisory Locks (Açık kilitler) |
| **Eşzamanlılık, Verim** | İndeksleme: Partial, Expressions, B-tree, Multicolumn  Gelişmiş İndeksleme: GiST, SP-Gist, KNN Gist, GIN, BRIN, Covering indexes, Bloom filters  Gelişmiş sorgu planlayıcı/iyileştirici, yalnızca dizin taramaları, çok sütunlu istatistikler  İşlemler, İç İçe İşlemler (kayıt noktaları aracılığıyla)  Multi-Version Concurrency Control (MVCC)  Okuma sorgularının paralelleştirilmesi ve B-ağacı dizinleri oluşturma  Tablo bölümleme  Serileştirilebilir de dahil olmak üzere SQL standardında tanımlanan tüm işlem yalıtım seviyeleri  Tam zamanında (JIT) ifadelerin derlenmesi |
| **Güvenilirlik,**  **Olağanüstü Durum Kurtarma** | Write-ahead Logging (WAL)  Çoğaltma: Eşzamansız, Eşzamanlı, Mantıksal  Anında kurtarma (PITR), aktif beklemeler  Tablo alanları |
| **Güvenlik** | Kimlik Doğrulama: GSSAPI, SSPI, LDAP, SCRAM-SHA-256, Sertifika vb.  Sağlam erişim kontrol sistemi  Sütun ve satır düzeyinde güvenlik  Sertifikalar ve ekstra bir teknik ile çok faktörlü kimlik doğrulama |
| **Genişletilebilirlik** | Saklı prosedürler ve işlevler  Prosedür Dilleri: Python, PL/PGSQL, Perl, vb.  SQL/JSON yol ifadeleri  Yabancı veri sarmalayıcılar: Standart SQL arayüzünden alternatif veri tabanlarına veya akışlara bağlanılabilir.  Tablolar için özelleştirilebilir depolama arayüzü  PostGIS ile birlikte ekstra pratiklik sağlayan birçok uzantı |
| **Uluslararasılaşma,**  **Metin arama** | Uluslararası karakter seti için destek  Büyük/küçük harfe duyarsız ve vurguya duyarsızlık  harmanlamalar  Tam metin arama |

TABLO ADI

http PROTOKOLÜ